

WHAT IS CLAIMED IS:

1 1. In a computer system, a method of remotely
2 monitoring execution of a computer program, comprising the
3 steps of:

4 modifying the computer program to include at least
5 one monitoring instruction;
6 executing the computer program;
7 the at least one monitoring instruction collecting
8 data regarding the execution of the computer program; and
9 sending the collected data to a remote system.

1 2. The method of claim 1, further comprising the
2 step of automatically sending the collected data to the remote
3 system when the computer program finishes execution.

1 3. The method of claim 1, further comprising the
2 step of changing the at least one monitoring instruction over
3 the computer program development cycle.

1 4. The method of claim 1, further comprising the
2 step of classifying the execution of the computer program as
3 normal or abnormal.

1 5. The method of claim 4, further comprising the
2 step of saving the call stack in the collected data if the
3 execution of the computer program is classified as abnormal.

1 6. The method of claim 1, further comprising the
2 step of allowing a user to customize processing that will be
3 performed when the computer program finishes execution.

1 7. The method of claim 1, further comprising the
2 step of generating a symbolic call stack on the remote system
3 (server) so that the computer program may be debugged remotely.

1 8. The method of claim 1, wherein the computer
2 program is compiled on the remote system (server) and further

3 comprising the step of storing a module map when the computer
4 program is compiled on the remote system.

1 9. The method of claim 8, further comprising the
2 step of storing a call stack and module list when the computer
3 program finishes execution.

1 10. The method of claim 9, further comprising the
2 step of generating a module name/RVA list from the call stack
3 and the module list.

1 11. The method of claim 10, further comprising the
2 step of sending the module name/RVA list to the remote system.

1 12. The method of claim 11, further comprising the
2 step of generating a symbolic call stack on the remote system
3 from the module map and the module name/RVA list so that the
4 computer program may be debugged remotely.

1 13. The method of claim 1, further comprising the
2 step of remotely debugging the computer program.

1 14. The method of claim 1, further comprising the
2 step of sending a version of the computer program to the remote
3 system during execution of the computer program.

1 15. The method of claim 14, further comprising the
2 step of downloading a new version of the computer program from
3 the remote system.

1 16. The method of claim 1, further comprising the
2 step of sending information to a bug tracking application.

1 17. The method of claim 1, further comprising the
2 step of, for each portion of the computer program designed by a
3 different vendor, collecting data specific to each portion.

1 18. The method of claim 1, wherein the at least one
2 monitoring instruction specifies a vendor.

1 19. The method of claim 1, further comprising the
2 step of adding the at least one monitoring instruction to
3 source code of the computer program.

1 20. The method of claim 19, further comprising the
2 step of utilizing a Windows hook to intercept a system call
3 invoked by the computer program.

1 21. The method of claim 1, further comprising the
2 step of augmenting object code of the computer program to
3 include the at least one monitoring instruction.

1 22. The method of claim 1, wherein the at least one
2 monitoring instructions are computer platform independent.

1 23. A distributed computer system, comprising:
2 a server computer;
3 a client computer in communication with the server
4 computer; and
5 a computer program running on the client computer
6 that includes at least one monitoring instruction that collects
7 and sends data regarding execution of the computer program to
8 the server computer.

1 24. The distributed computer system of claim 23,
2 further comprising a transport medium connecting the client
3 computer and the server computer.

1 25. The distributed computer system of claim 24,
2 wherein the transport medium is a network or media.

1 26. The distributed computer system of claim 23,
2 further comprising a DLL for intercepting system calls.

1 27. The distributed computer system of claim 23,
2 further comprising a bug tracking application.

1 28. The distributed computer system of claim 23,
2 further comprising an expansion mechanism for augmenting the
3 computer program to include the at least one monitoring
4 instruction.

1 29. A computer program product for remotely
2 monitoring execution of a computer program, comprising:
3 a computer readable storage medium storing the
4 computer program comprising:
5 code that calls at least one monitoring instruction,
6 the at least one monitoring instruction collecting data
7 regarding the execution of the computer program;
8 and code that sends the collected data to a remote
9 system.

1 30. The computer program product of claim 29,
2 further comprising code that automatically sends the collected
3 data to the remote system when the computer program finishes
4 execution.

1 31. The computer program product of claim 29,
2 further comprising code that classifies the execution of the
3 computer program as normal or abnormal.

1 32. The computer program product of claim 29,
2 further comprising code that saves the call stack in the
3 collected data if the execution of the computer program is
4 classified as abnormal.

1 33. The computer program product of claim 29,
2 further comprising code that allows a user to customize
3 processing that will be performed when the computer program
4 finishes execution.

1 34. The computer program product of claim 29,
2 further comprising code that stores a call stack and module
3 list when the computer program finishes execution.

1 35. The computer program product of claim 34,
2 further comprising code that generates a module name/RVA list
3 from the call stack and the module list.

1 36. The computer program product of claim 35,
2 further comprising code that sends the module name/RVA list to
3 the remote system.

1 37. The computer program product of claim 29,
2 further comprising code that downloads a version of the
3 computer program from the remote system.

1 38. The computer program product of claim 29,
2 further comprising code that sends information to a bug
3 tracking application.

1 39. The computer program product of claim 29,
2 wherein the at least one monitoring instruction specifies a
3 vendor.